



<http://www.android.com/>

Hybrid apps with HTML5
Android NDK

Web tech on the mobile



- HTML5 currently under development
- New syntactical features. These include the `<video>`, `<audio>` and `<canvas>` elements
- Integration of Scalable Vector Graphics, SVG, content
- Drag-and-drop
- Full-scale application development for the browser possible
- Apps might be platform agnostic (unaware of host env.)
- Android comes with a HTML5 Webkit-based browser
- Dive into HTML5
 - <http://diveintohtml5.info/>

Web tech on the mobile



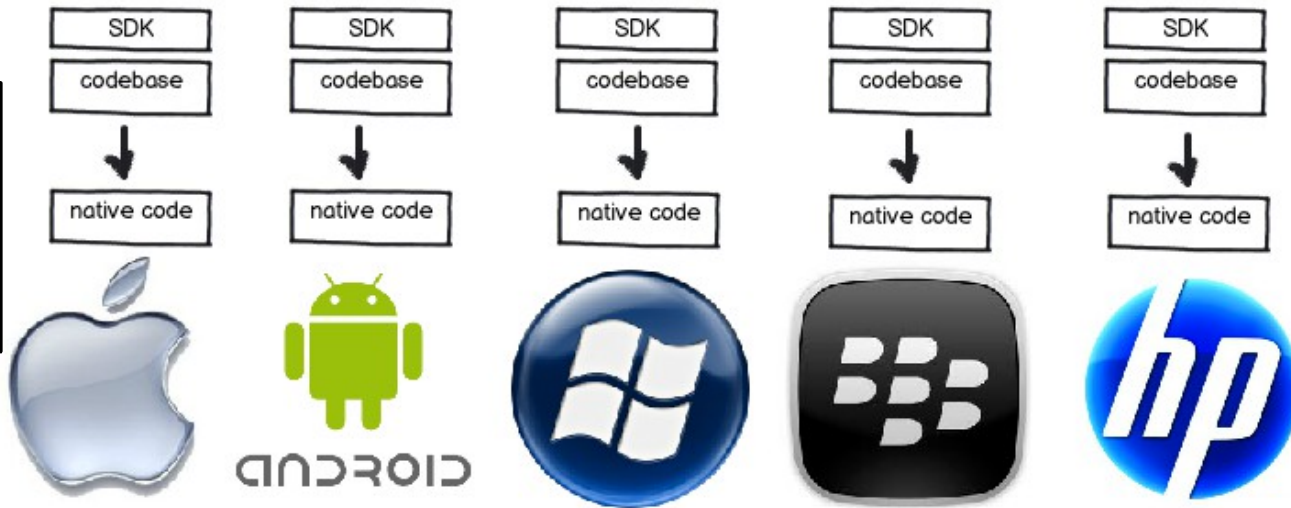
- **Pamela Fox's talks – many presentations about hybrid solutions**
 - <http://www.pamelafox.org/talks>
 - PhoneGap: Why & How – presentation from 2012-06
- **Mobile Developers Guide To The Galaxy**
 - The "Cross-Platform App Frameworks" chapter
- HTML5 could eventually become the platform that all apps, whether for mobiles or for PCs, are written on. "It's taken 20 years because the underlying standards had to evolve from the proprietary Mac and Windows platforms. But they have both adopted HTML5 as a future direction. So that means in future a lot of apps will be running in HTML5 in both mobile and non-mobile form." - ES



Google CEO Eric Schmidt speaks at the Mobile World Conference in Barcelona.

Native app

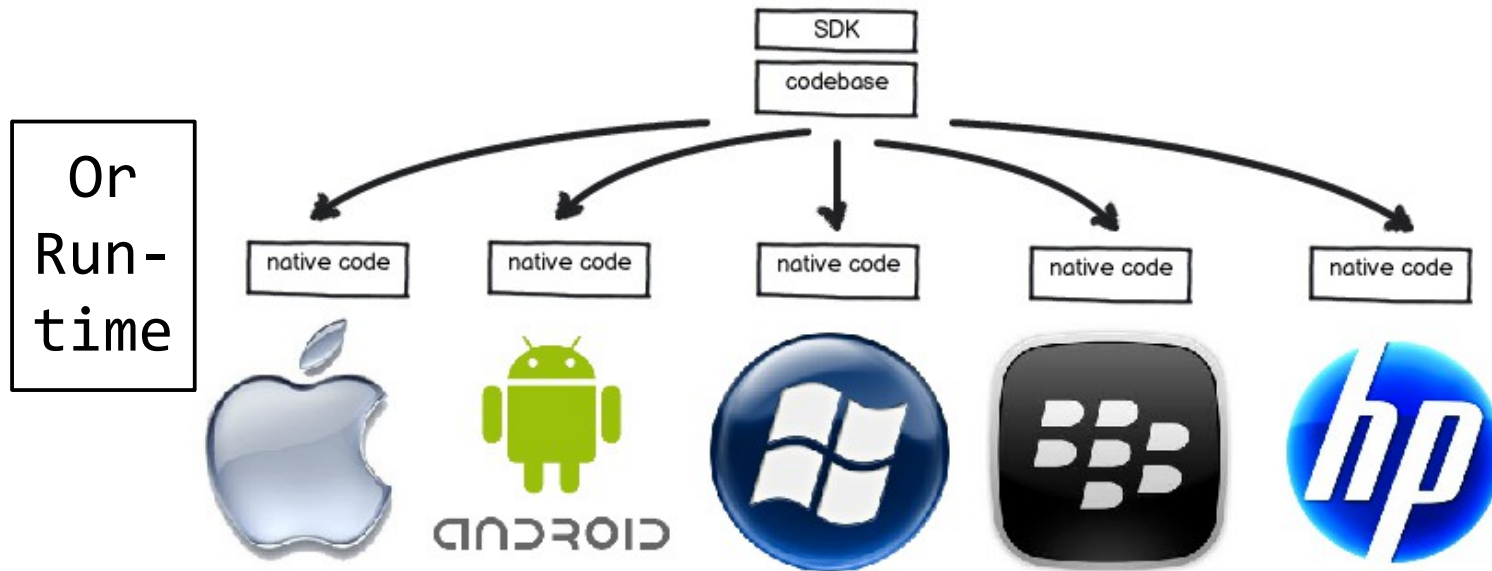
Or
Run-
time



Platform	Language	IDE
iOS	Objective C	XCode
Android	Java	Eclipse
Windows	.NET	Visual Studio
Blackberry	Java	BB JDE
hpWebOS	C/C++/HTML5	?

Platforms: **1** Perf: **Best** UX: **Best** APIs: **All**

Cross-Compiled code



			iOS	Android	WP7	BB	WebOS
Titanium	JavaScript	0-\$500-\$\$/Apache	Green	Green	Red	Red	Red
ParticleCode	Java, AS3	Free	Green	Green	Green	Green	Green
XMLVM	Java, .NET	Free/LGPL	Green	Green	Red	Red	Red
Monotouch	C#, .NET	\$400-\$3000/OS	Green	Green	Red	Red	Red

Platforms: **Multi** Perf: **Great** UX: **Great** APIs: **All**

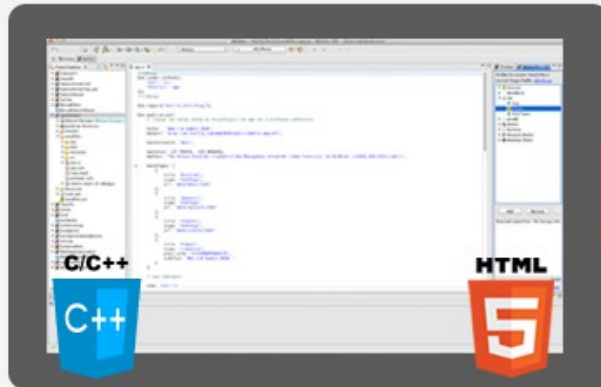
Swedish cross compiler tool

MoSync SDK

<http://www.mosync.com>

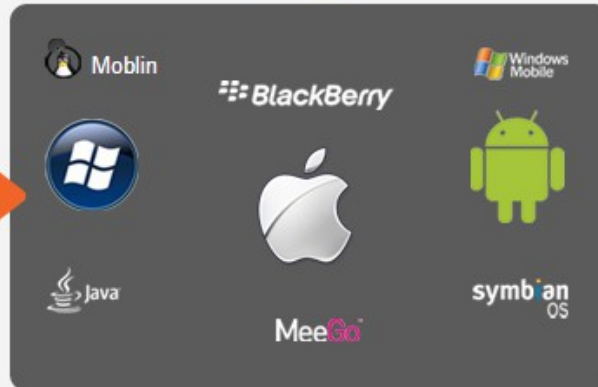
MoSync SDK - native mobile app development for multiple platforms using a single code base

The open-source MoSync Software Development Kit (SDK) is a rich cross-platform mobile application development environment that makes it easy to develop apps for all major mobile platforms from a single code base. The SDK enables mobile developers to build and compile apps for up to nine different platforms at once, using C/C++ or HTML5/JavaScript, or a combination of both to create hybrid apps.



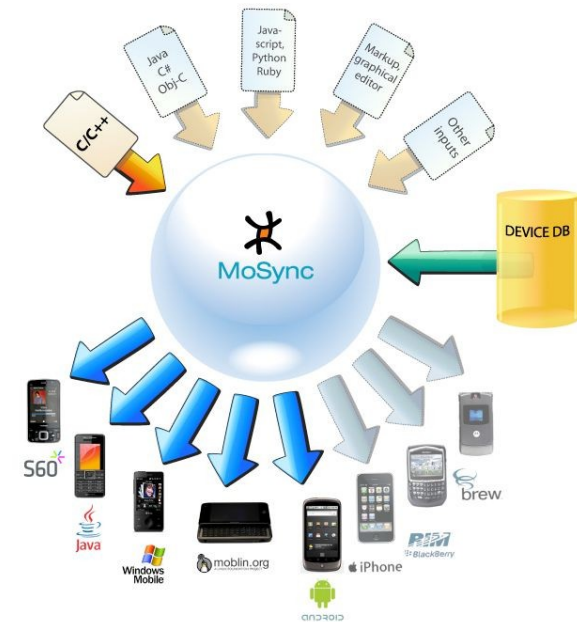
Develop in C/C++ or HTML5

The MoSync SDK is a rich cross-platform IDE with a C/C++ foundation and tight integration with MoSync Reload, which allows you to easily create hybrid apps using also HTML5/JavaScript.



Deploy to Multiple Platforms

The SDK enables you to compile your app for up to nine different platforms, using one single code base. You can then focus on your next app idea.



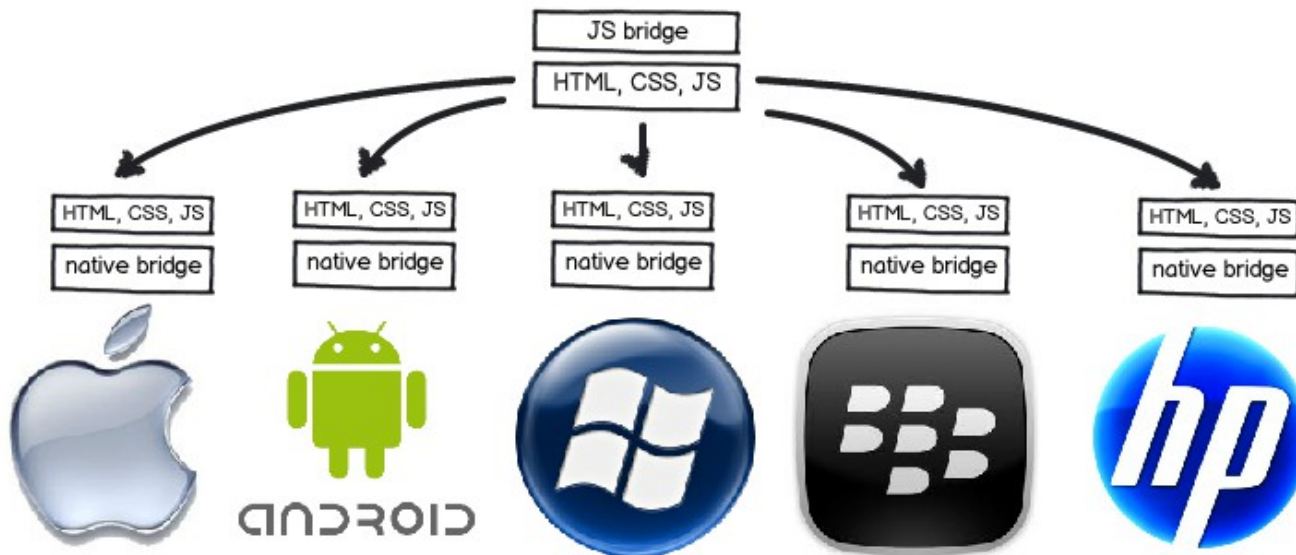
Download

Learn More

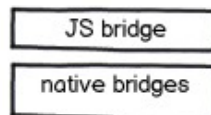


Watch Mosync SDK Video
MoSync beginner's tutorial of how to start the SDK, import examples, run the MoSync emulator and Bluetooth export to a mobile phone.

Hybrid app

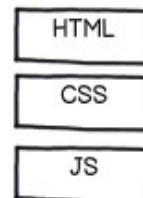


Bridge SDK



+

Web Framework



= Hybrid App

Platforms: **Multi** Perf: **Subpar** UX: **Subpar** APIs: **Many**

Bridge SDK



APACHE
CORDOVA™

		iOS	Android	WP7	BB	WebOS
(Adobe) PhoneGap	Free/Apache	Green	Green	Green	Green	Green
appMobi	Free/MIT	Green	Green	Red	Red	Red
trigger.io	\$600-\$3000	Green	Green	Red	Red	Red

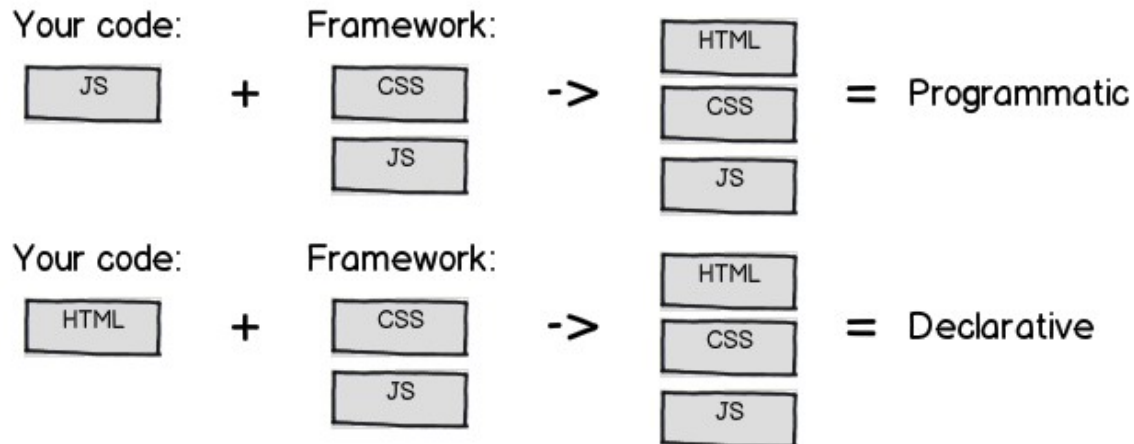
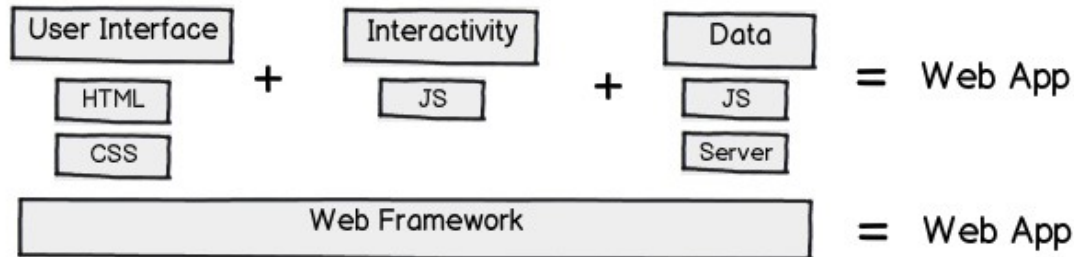
```

<!DOCTYPE html>
<html>
  <head>
    <script type="text/javascript" charset="utf-8" src="phonegap.js"></script>
    <script type="text/javascript" charset="utf-8">
      function capturePhoto() {
        navigator.camera.getPicture (onPhotoDataSuccess);
      }

      function onPhotoDataSuccess(imageData) {
        var smallImage = document.getElementById('smallImage');
        smallImage.src = "data:image/jpeg;base64," + imageData;
      }
    </script>
  </head>
  <body>
    <button onclick="capturePhoto();">Capture Photo</button> <br>
  </body>
</html>

```

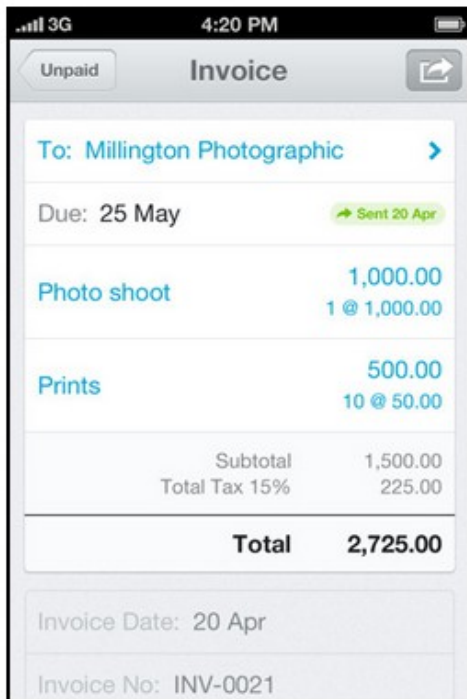

Web Frameworks



- Declarative programming is
 - A program that describes what computation should be performed and not how to compute it
 - Any programming language that lacks side effects (or more specifically, is referentially transparent)
 - A language with a clear correspondence to mathematical logic as SQL, regular expressions, logic programming, functional programming, etc.

Programmatic Frameworks

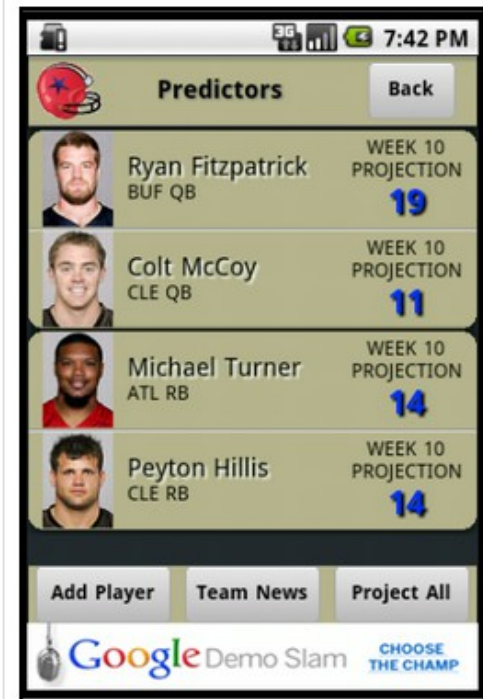
Sencha



Jo

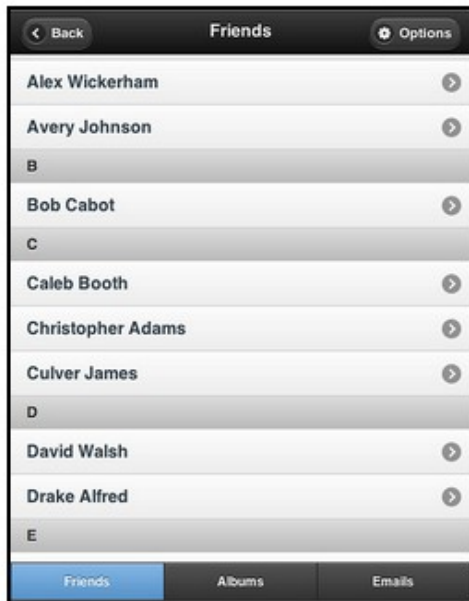


GWTRMobile



Declarative Frameworks

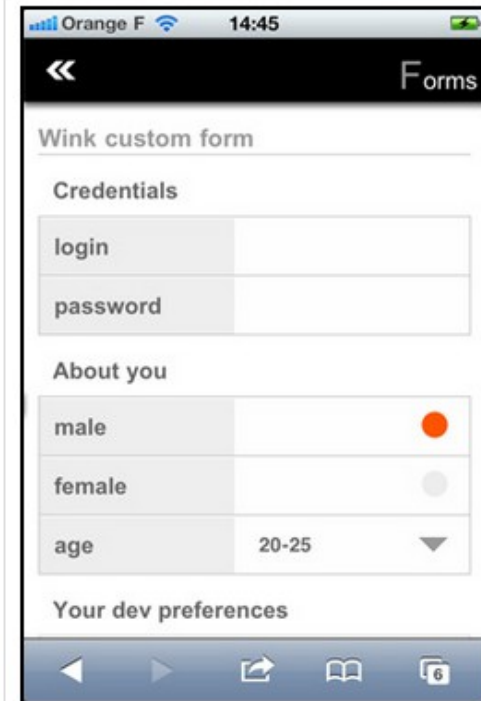
jQueryMobile



Kendo UI



Wink



CSS Frameworks

[Twitter Bootstrap](#)

Bootstrap
☰

Basic form

With v2.0, we have lighter and smarter defaults for form styles. No extra markup, just form controls.

Label name

Associated help text!

Check me out

Search form

Reflecting default WebKit styles, just add `.form-search` for extra rounded search fields.

[ZURB Foundation](#)

swizzle
MENU

Home
Features
Chat with us → Hi!

The most amazing **all-in-one** system.

You'll be wondering where we've been all your life ;)

swizzle.com/Admin/Frameset.aspx

Template: v01-Brochure

More Options

Page Content

it's got heart

We take pride in offering first-class customer ser

My Panel

ing Page: dashboard

Membership Access:

Membership Cost: UNITED STATES

Currency:

Membership Cost (\$): 39.00

Membership Renewal: Monthly

Once Daily

JS Libraries

	DOM	Data	AJAX	Touch	Scroll	Animation
Zepto	Green	Red	Green	Green	Green	Green
jQuery	Red	Red	Red	Green	Green	Green
iScroll	Red	Red	Red	Red	Green	Red
Backbone	Red	Green	Red	Red	Red	Red

microjs

I need ... ✓

- a base framework.
- a DOM utility.
- templating.
- CSS animation.
- JavaScript animation.
- a CSS selector engine.
- data manipulation.

Zepto Example

```
var activate = ('createTouch' in document) ? 'touchstart' : 'click';

$("body > section").first().addClass("current")

$("a.back").live(activate, function(event) {
  var current = $(this).attr("href");
  $(".current").removeClass("current").addClass("reverse");
  $(current).addClass("current");
});

$(".menu a[href]").live(activate, function(event) {
  var link = $(this), section = link.closest('section'),
      prev_element = "#"+(section.removeClass("current").addClass("reverse").attr(
  $(link.attr("href")).addClass("current");
  $(".current .back").remove();
  $(".current .toolbar").prepend("Back");
});
```

<http://jq tj s.com/preview/demos/todo/>

[Try full demo](#)

My app stack

APACHE
CORDOVA™

PhoneGap

+

Twitter

+

Zepto JS

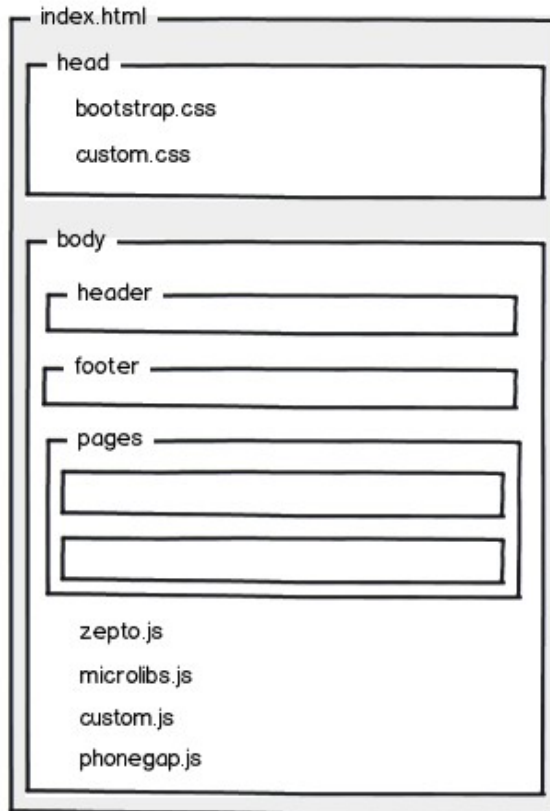
Bootstrap

The screenshot displays two views of the 'eatdifferent' mobile application. The left view is a data entry screen for Wednesday, Sep. 14, 2011, featuring sections for 'Measurements' (Weight: 150 lbs), 'Meals' (Omelette at 9 AM and Turkey sandwich & wine at 5:30 PM), and 'Food Goals' (sliders for Added Sugars, Grains, Dairy, Fruits, Alcohol, and Processed Food). The right view is a dashboard for Thursday, Dec. 22, 2011, with a 'Quick Pic' button and a list of menu items: Measurements, Meals, Food Goals, Food Bonuses, Tags, and Notes, along with a 'Feedback' button and a bottom navigation bar with 'Log', 'Stream', and 'Logout' options.

<http://www.eatdifferent.com/>



PhoneGap: Architecture





PhoneGap: index.html

```
<!doctype html>
<html>
  <head>
    <meta name="viewport" content="initial-scale=1.0, user-scalable=no, width=device-w
    <link rel="stylesheet" href="css/all-phonegap-min.css"/>
  </head>
  <body>

    <div id="mobile-header" class="actionsbar">
      
    </div>

    <div id="mobile-footer" class="navbar">
      <ul class="nav">
        <li class="active"><a href="#mobile-log-page">Log</a></li>
        <li><a href="#mobile-stream-page">Stream</a></li>
        <li><a id="logout-button">Logout</a></li>
      </ul>
    </div>

    <!-- Pages -->
    <div id="mobile-pages" class="container-fluid">

      <div id="mobile-splash-page" class="mobile-page">
        Loading...
      </div>

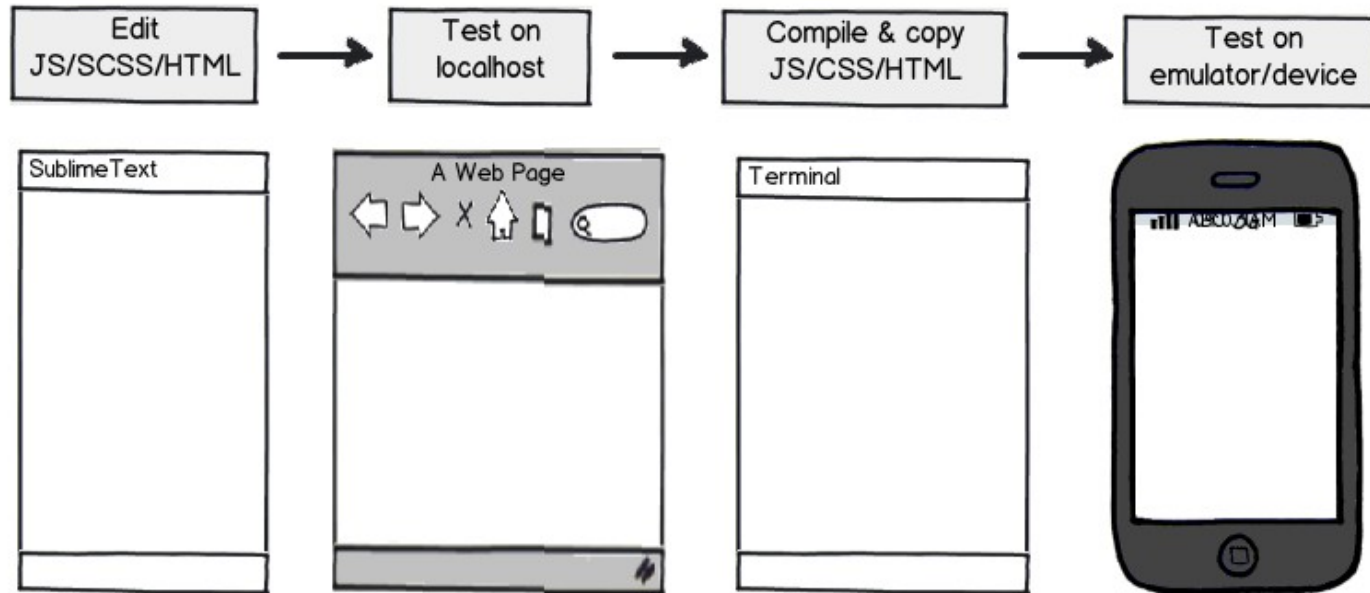
      <div id="mobile-login-page" class="mobile-page">
        <p>Welcome! Please login with either your Facebook account or email.</p>
      </div>
    </div>
```

<https://gist.github.com/pamelafox/1515763>

[Gist: PhoneGap index.html](https://gist.github.com/pamelafox/1515763)



PhoneGap: Workflow



Tools: [SASS](#), [Compass](#), [Resolutionizer](#), [YUI CSS Compressor](#), [JSHint](#), [Closure Compiler](#), [Jinja2](#), [Android SDK](#), [XCode](#)



PhoneGap: Debugging

Android: [DDMS](#)

	Application	Tag	Text
PhoneGap	io.everyday.app	PhoneGapLog	LOG: Setting current date
Web Console	io.everyday.app	PhoneGapLog	LOG: Fetching URL: https://android-dot-everyday-app.appspot.
	io.everyday.app	PhoneGapLog	file:///android_asset/www/js/all-phonegap-min-04201555.js: L
	io.everyday.app	PhoneGapLog	LOG: Fetching URL: https://android-dot-everyday-app.appspot.
	io.everyday.app	PhoneGapLog	LOG: textStatus: error errorThrown: null status:0
	io.everyday.app	PhoneGapLog	file:///android_asset/www/js/all-phonegap-min-04201555.js: L
	io.everyday.app	PhoneGapLog	LOG: textStatus: error errorThrown: null status:0
	io.everyday.app	PhoneGapLog	LOG: Fetching the stream for real for: 7everyone

iPhone: [XCode Debug Log](#)

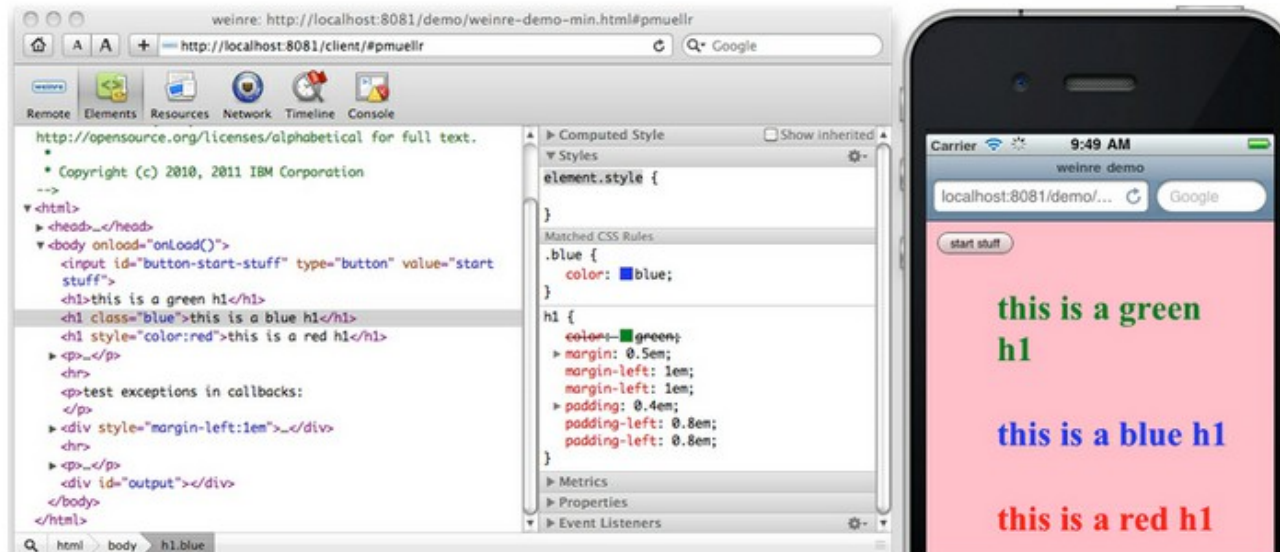
```
function log(something) {
  // ...
  if (isIOS() || isAndroid()) {
    if (typeof something == 'object') {
      something = JSON.stringify(something);
    }
    something = truncateText(something, 2000);
    something = '\nLOG: ' + something;
    var stacktrace = '';
    if (window.printStackTrace) {
      try {
        stacktrace = '\n -' + printStackTrace().slice(4).join('\n -');
        something += '\nSTACKTRACE:' + stacktrace;
      } catch(e) {}
    }
  }
}
```

PhoneGap: Debugging DOM



In development:

```
<script src="http://debug.phonegap.com/target/target-script-min.js#eatdifferent">
</script>
```



Tools: [Weinre](#), [debug.phonegap.com](#), [Adobe Shadow](#)

Cordova platform support

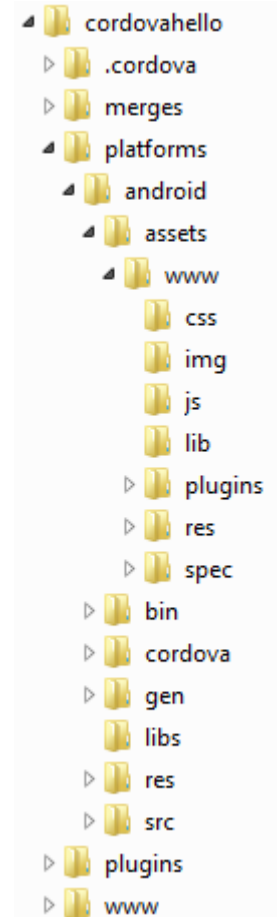
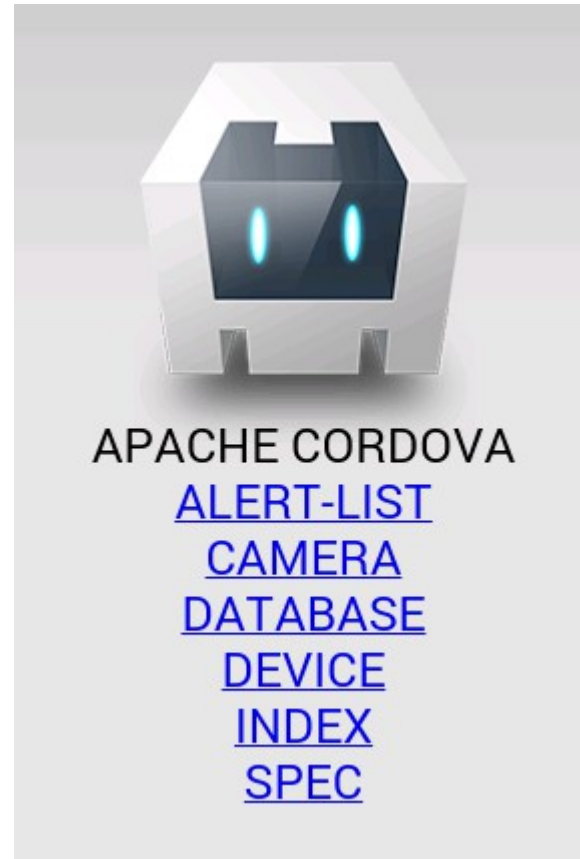


	android	blackberry (6)	blackberry10	ios	wp7 (Windows Phone 7)	wp8 (Windows Phone 8)	win8 (Windows 8)	firefoxos	tizen
cordova CLI	✓ Mac, Windows, Linux	✗ Mac, Windows	✓ Mac, Windows	✓ Mac	✓ Windows	✓ Windows	✗	✗	
Embedded WebView	✓ (see details)	✗	✗	✓ (see details)	✗	✗	✗	✗	
Plug-in Interface	✓ (see details)	✓ (see details)	✓ (see details)	✓ (see details)	✓ (see details)	✓	✗	✗	
Platform APIs									
Accelerometer	✓	✓	✓	✓	✓	✓	✓	✓	✓
Camera	✓	✓	✓	✓	✓	✓	✓	✓	✓
Capture	✓	✓	✓	✓	✓	✓	✗	✗	
Compass	✓	✗	✓	✓ (3GS+)	✓	✓	✓	✓	✓
Connection	✓	✓	✓	✓	✓	✓	✓	✓	✓
Contacts	✓	✓	✓	✓	✓	✓	✓	✗	
Device	✓	✓	✓	✓	✓	✓	✓	✓	✓
Events	✓	✓	✓	✓	✓	✓	✓	✓	✓
File	✓	✓	✓	✓	no FileTransfer	no FileTransfer	✓	✗	
Geolocation	✓	✓	✓	✓	✓	✓	✓	✓	✓
Globalization	✓	✓	✗	✓	✗	✓	✗	✗	
InAppBrowser	✓	✓	✓	✓	✓	✓	✗	✗	
Media	✓	✗	✓	✓	✓	✓	✓	✓	✓
Notification	✓	✓	✓	✓	✓	✓	✓	✓	✓
Splashscreen	✓	✗	✓	✓	✓	✓	✓	✗	
Storage	✓	✓	✓	✓	localStorage only	localStorage only	✓	✓	

Cordova demo



- Demo: CordovaHello
- NODE.JS
- CLI driven
- Plugins
- Eclipse
 - assets/www/**
 - res/xml/config.xml
 - AndroidManifest.xml
- HTML
- CSS
- JavaScript
- Instructions in lab



– http://cordova.apache.org/docs/en/3.1.0/guide_cli_index.md.html

Conclusion

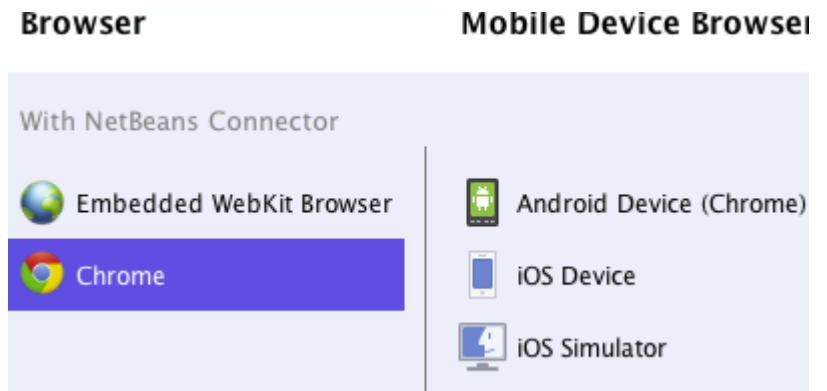


- PROs
 - Code reuse
 - Knowledge reuse
 - Bright future
- CONs
 - Mobile browser bugginess/stability
 - Bugs take longer to debug
 - Hard to fake native UX
- Compatibility tables for support of HTML5, CSS3, SVG and more in desktop and mobile browsers
 - <http://caniuse.com/>
- Performance guidelines
 - <http://developer.yahoo.com/performance/>
 - <https://developers.google.com/speed/>

Web tech on the mobile



- NetBeans IDE 7.4 Release Candidate 1 Information
 - NetBeans IDE 7.4 extends the advanced HTML5 development support introduced in NetBeans IDE 7.3 to Java EE and PHP application development, while offering new support for hybrid HTML5 application development (Cordova) on the Android and iOS platforms.
 - <https://netbeans.org/community/releases/74/>
- Webbtekniken räcker inte till på mobilen
 - <http://computersweden.idg.se/2.2683/1.523834/webbtekniken-racker-inte-till-pa-mobilen>

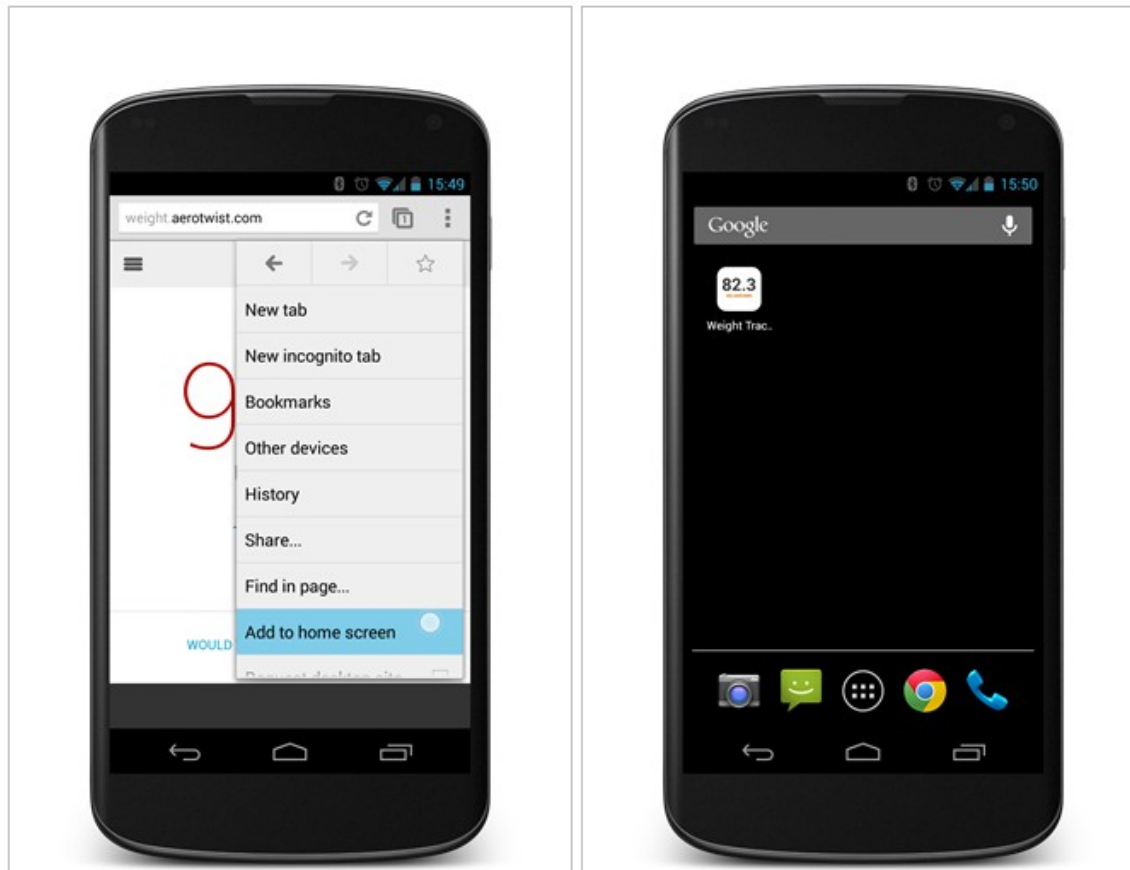


Web apps



- **Google Chrome Beta 31 For Android Brings 'Install To Homescreen' For Web Apps With Support For Full-Screen Mode And Task-Switcher Integration**

Starting with the upcoming version 31, Chrome for Android Beta will support adding web apps to the home screen via an option in the overflow menu. These apps will still run in a sandboxed Chrome environment, but they will also be integrated with the operating system in a few interesting ways.

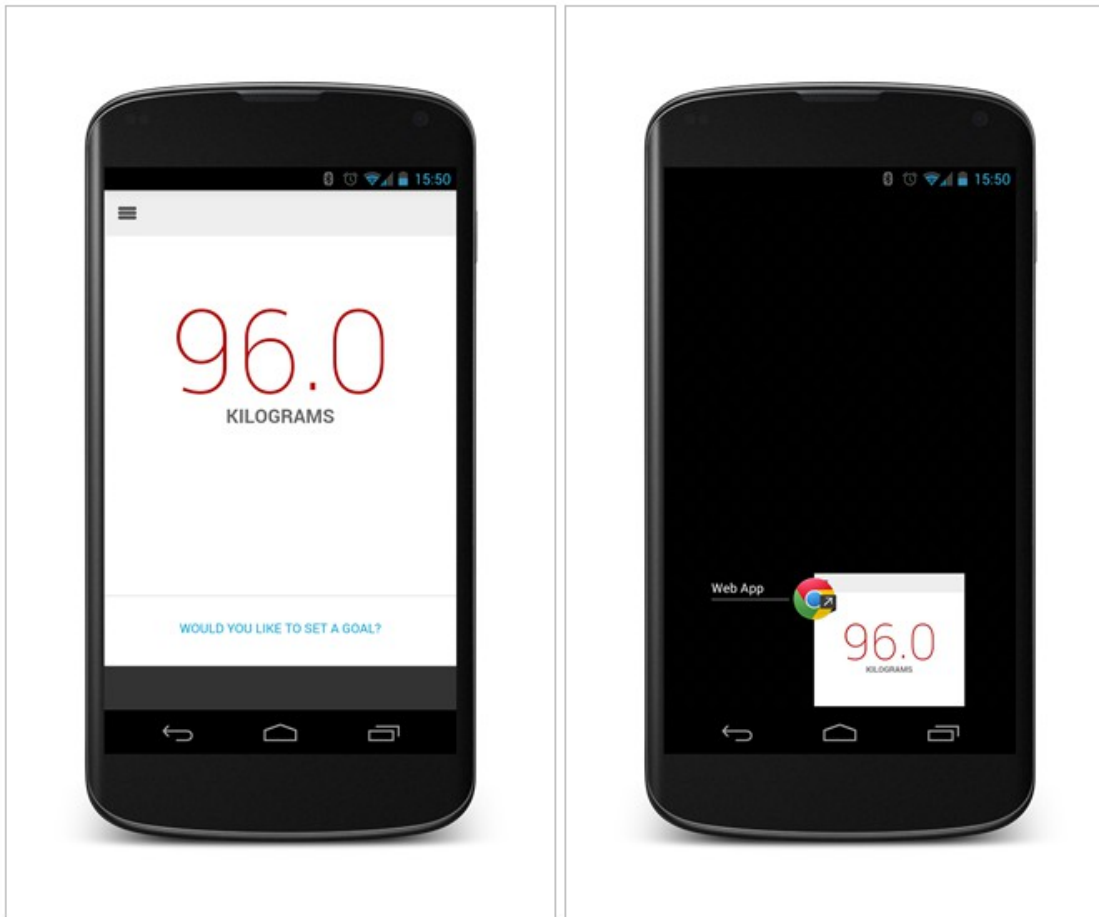


<https://developers.google.com/chrome/mobile/docs/installtohomescreen>

Web apps



Adding a supported web app to the homescreen will place a custom icon on the device, but this is more than a bookmark shortcut. The icon may look like a regular app icon, but tapping it brings up what is essentially a full-screen Chrome frame. You can also switch between web apps just like a native app (i.e. they are not lumped in with Chrome). The device will also alert users if the web app tries to send them to another domain.



Developers can easily add the necessary tags to have their web apps supported by the new functionality. Chrome will also recognize iOS Safari's Add to Homescreen tag, but only for the time being – support for the "apple-mobile-web-app-capable" meta tag will be dropped in a later release.

Native, Web App or Hybrid?



Device features - Access to device-specific features as sensors, camera, gestures, notifications, etc.

Offline functioning - In-browser caching is available in HTML5, but it's limited.

Discoverability - Web apps win the prize on discoverability.

Speed - Native apps win the speed competition. In 2012 Mark Zuckerberg declared that Facebook's biggest mistake had been betting on the mobile web and not going native.

Installation - Installing a native or hybrid app is a hassle for users.

Maintenance - Maintaining a native (multi-platform) app can be complicated. While maintaining a web app or a hybrid app can be done as often or as frequently as needed.

Platform independence - You definitely have a better chance of achieving it with web apps and hybrid apps than with native apps.

Content restrictions - Approval process, and fees etc. in app stores.

Development cost - It's arguably cheaper to develop hybrid and web apps.

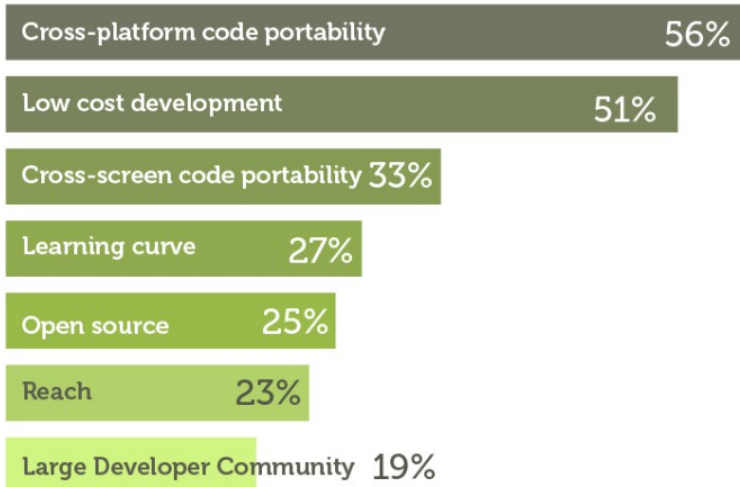
User Interface - Native apps are the way to go.

To summarize - Mobile apps cater to the needs of the mobile user. There is no unique best solution: each of these has their strengths and weaknesses. The choice of one versus the other depends on each company's unique needs.

HTML5 trades off native optimisation for portability & cost

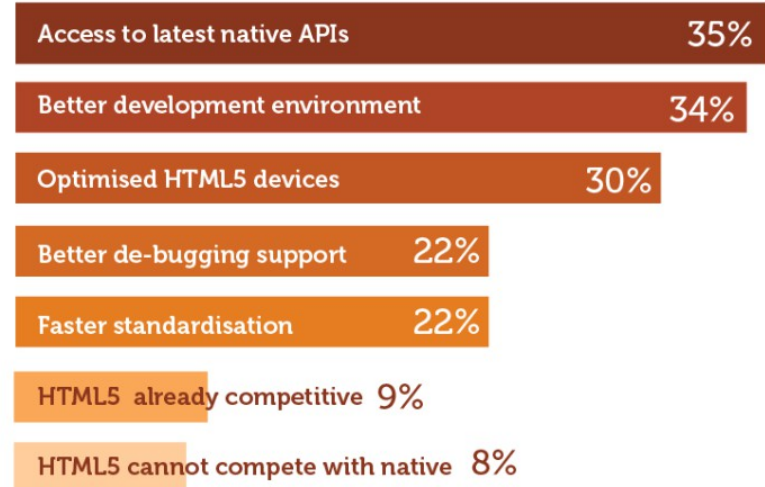
Reasons for selecting HTML5

% of developers selecting each options among those who use or plan to adopt HTML5 (n=1974, weighted)



What HTML5 needs to compete with native

% of developers selecting each options among those who use or plan to adopt HTML5 (n=2063, weighted)



JavaScript



- Jeff Atwood mainly known for the programming blog Coding Horror, and as the co-founder of the question-and-answer website Stack Overflow and the Stack Exchange Network.
- Atwood's Law: any application that can be written in JavaScript, will eventually be written in JavaScript. ... (whether or not it is suitable)
- Tim Berners-Lee

The Principle of Least Power

July 17, 2007

Tim Berners-Lee on [the Principle of Least Power](#):

Computer Science spent the last forty years making languages which were as powerful as possible. Nowadays we have to appreciate the reasons for **picking not the most powerful solution but the *least* powerful**. The less powerful the language, the more you can do with the data stored in that language. If you write it in a simple declarative form, anyone can write a program to analyze it. If, for example, a web page with weather data has RDF describing that data, a user can retrieve it as a table, perhaps average it, plot it, deduce things from it in combination with other information. At the other end of the scale is the weather information portrayed by the cunning Java applet. While this might allow a very cool user interface, it cannot be analyzed at all. The search engine finding the page will have no idea of what the data is or what it is about. The only way to find out what a Java applet means is to set it running in front of a person.

Android native components



- Android NDK, is not hard to install or to use (limited C++ support)
- It have a rather limited API and is intended for performance with
 - OpenGL, including support for some newer versions that the (Java) SDK supports
 - Math (some, but not all, calculation-intensive algorithms might benefit from being done on the native layer)
 - 2D graphics – pixelbuffer support (only starting with 2.2)
 - libc – it's there for compatibility and perhaps to allow you to port existing native code
- You need to install C/C++ support in Eclipse mainly for syntax coloring and checking
- In Windows you need Cygwin 1.7.x and all devel branch packages
- The Android NDK

<http://developer.android.com/tools/sdk/ndk/index.html>

Using the NDK with Java Native Interface



- The idea is to put your native pieces of code into libraries/modules that you can then consume from the Java code via JNI

```
// C/C++ files are placed in the <project>/jni/ folder
#include <string.h>
#include <jni.h>

jint factorial(jint n){
    if(n == 1){
        return 1;
    }
    return factorial(n-1) * n;
}
// Java + package name + activity + function
jint Java_com_cookbook_advance_ndk_ndkact_factorial(
    JNIEnv* env, jobject javaThis, jint n ) {
    return factorial(n);
}
```

Type mapping Java <-> Native

Java Type in C/C++	Native Type	Description
jboolean	unsigned char	unsigned 8 bits
jbyte	signed char	signed 8 bits
jchar	unsigned short	unsigned 16 bits
jshort	short	signed 16 bits
jint	long	signed 32 bits
jfloat	float	32 bits
jlong	long long _int64	signed 64 bits
jdouble	double	64 bits

```
public class ndkact extends Activity {
    // loading the library/module - must match jni/android.mk
    static {
        System.loadLibrary("ndkcookbook");
    }
    // declaring the native function - must match cookbook.c
    private static native int factorial(int n);

    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        TextView tv = new TextView(this);
        tv.setText(" native calculation on factorial :"+factorial(30));
        setContentView(tv);
    }
}
```

```
# the <project>/jni/Android.mk file
# a make file needed to build the lib
#
LOCAL_PATH := $(call my-dir)
include $(CLEAR_VARS)
LOCAL_MODULE := ndkcookbook
LOCAL_SRC_FILES := cookbook.c
include $(BUILD_SHARED_LIBRARY)
```

Lab 6 review and next lecture

- Lab 6 got two tasks
 - XML/JSON parsing and view in a nice GUI
 - <http://apikatalogen.se/>
 - Hybrid application
- You must be able or know how to
 - understand all the previous points from former labs
 - use Loaders
 - use the Apache HTTP classes
 - use JSON and XML
 - know how hybrid applications work
 - know how SQLite databases and content providers work
- Next last lecture?
 - Graphics, security, repetition or sleep?